IBM DB2 Analytics Accelerator for z/OS
Version 5.1.0

# *Encryption of Data in Motion*

SH12-7077-01

**Note**

Before you use this information and the product it supports, read the information in "Notices" on page 47.

**Second Edition, May 2016**

This edition applies to version 5.1.0 of IBM DB2 Analytics Accelerator for z/OS (product number 5697-DA5), and to all subsequent releases and modifications until otherwise indicated in new editions. This edition replaces SH12-7077-00. Changes to this edition are marked with a vertical bar.

# Contents

# Figures

# Enabling encryption of data in motion

Earlier versions of IBM® DB2® Analytics Accelerator for z/OS® were delivered without a network encryption function. Even though this did not pose a security risk when the setup recommendations were followed (dedicated private network between the mainframe computer or LPAR and the accelerator), it soon became obvious that in many situations, this type of setup did not align well with existing network infrastructures. Hence a secure way of data traffic had to be provided for customers who want to route sensitive data, such as patient data, credit-card transactions, or social security numbers through their corporate intranet. Furthermore, the security standards of many organizations demand that any sensitive data that is sent across a network must be encrypted. So finally, encryption capabilities were added to the product.

Naturally, the use of encryption comes at a price, and that is a slower performance, which is due to the fact that data has to be encrypted before it enters the network and decrypted after it leaves it. Both processes require a considerable amount of time, and the data volumes that are handled by IBM DB2 Analytics Accelerator for z/OS are usually extremely high. Customers rightfully expect remarkable acceleration rates despite the use of encryption. However, such rates can only be guaranteed in connection with fast hardware, which is why encryption is offered only with the IBM PureData® System for Analytics N2001 and N3001 series. For more information, see the IBM DB2 Analytics Accelerator prerequisites website.

To reduce the CPU consumption on the mainframe, IBM recommends the use of z Systems™ Integrated Information Processors (zIIPs) for IPsec processing. However, despite faster and specialized hardware, there will be a noticeable performance impact on bulk transmissions of data, such as table load jobs or queries with huge result sets. Queries with small or moderate result sets, on the other hand, will not be impacted by the use of encryption.

## Encryption solution - summary of features

- AES-GCM symmetric encryption for the network payload
- RSA 2048 bit encryption keys
- Public key certificate signed by shared certificate authority, type X.509 in PKCS#12 format

The following figure shows the components that are involved when you set up an encrypted network with the z/OS Communications Server. Some of the components with a yellow background must be configured for IPsec network encryption with IBM DB2 Analytics Accelerator for z/OS. For an in-depth discussion, see *Chapter 4. Policy Agent* in *IBM z™/OS V2R1 Communications Server TCP/IP Implementation Volume 4: Security and Policy-Based Networking*. You find a link to this Redbook at the end of this topic.

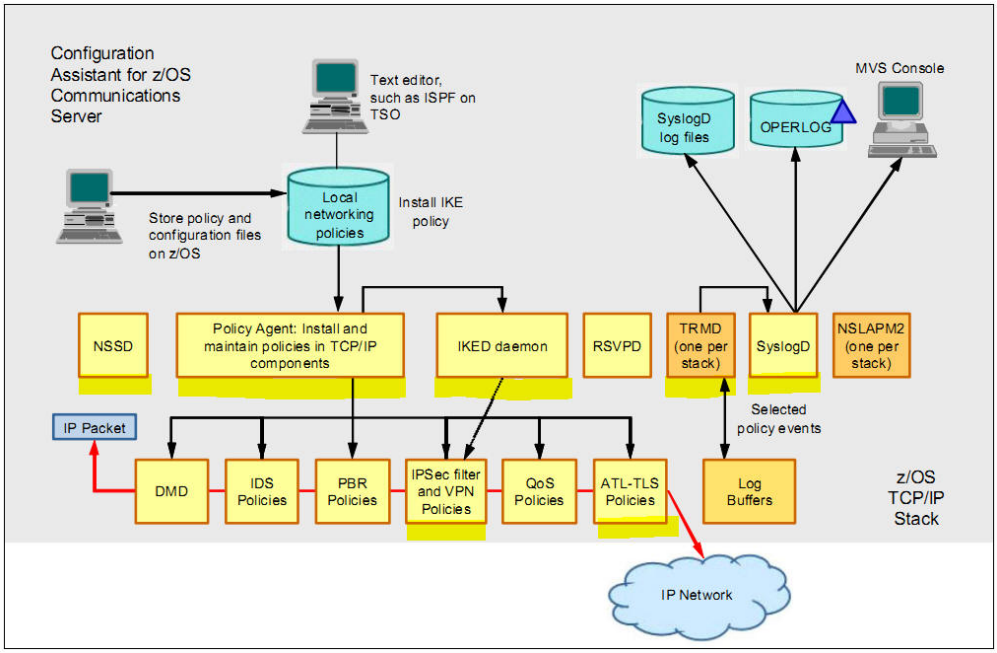


*Figure 1. Components in the IPsec setup with z/OS Communications Server*

If you want to encrypt the network traffic between a z/OS LPAR and an accelerator, you need an RSA key pair and a public key certificate that is signed by a shared certificate authority on each side for each LPAR or accelerator (communication endpoints). The following figure shows three LPARs that are connected to two accelerators.

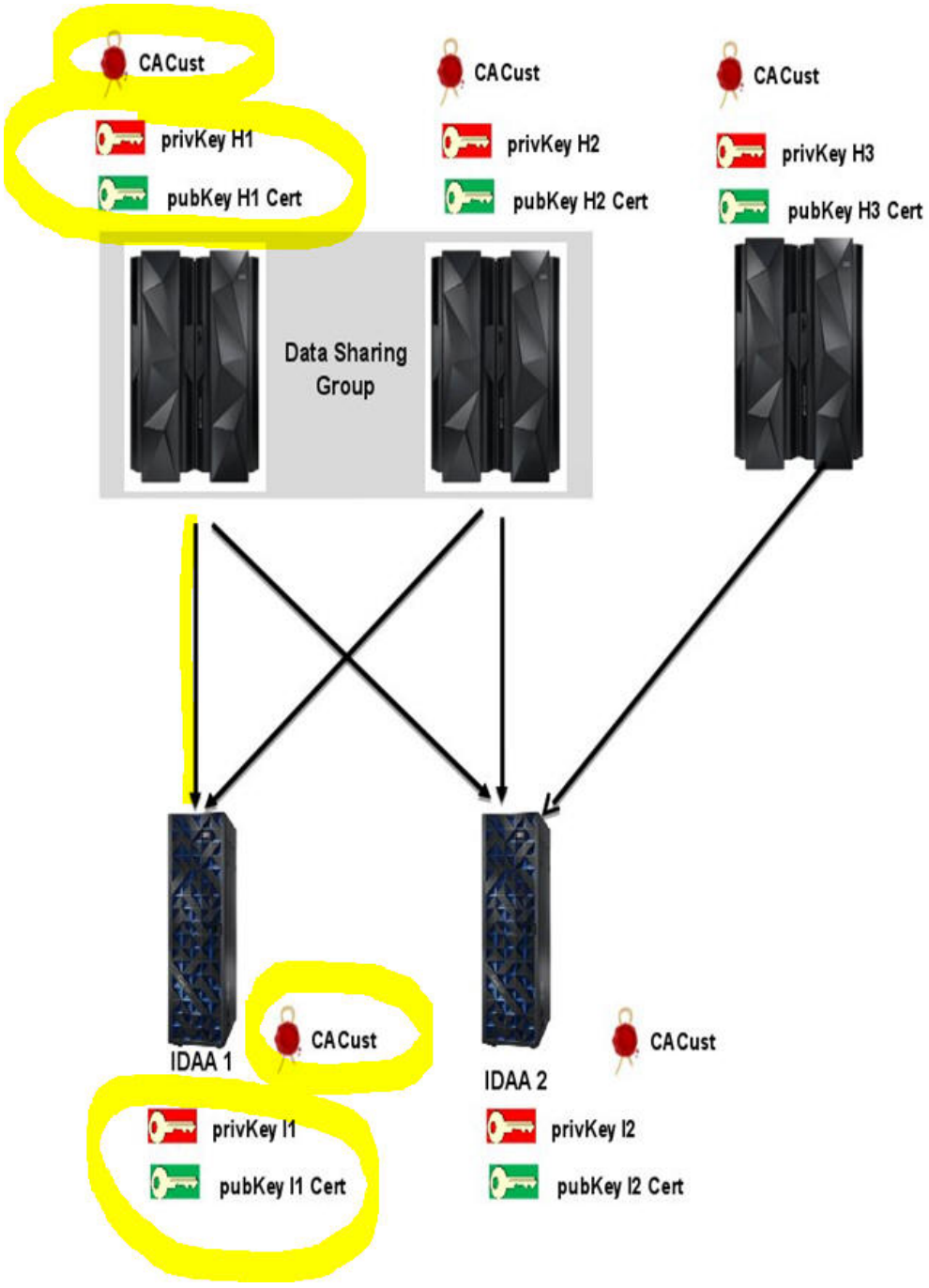


*Figure 2. Distribution of certificates and keys*

Each peer uses its IKE daemon to authenticate itself and negotiate the traffic protocol. It is your responsibility to generate the key pairs, sign them with the same certificate authority and then deploy and configure the keys with their associated certificates on the accelerators and z/OS LPARs. The following sections describe how to configure one connection from one LPAR to one accelerator (circled yellow in the previous figure).

**Related information**:

IBM z Systems Integrated Information Processor (zIIP)

 IBM z/OS V2R1 Communications Server TCP/IP Implementation Volume 4: Security and Policy-Based Networking

# Installing prerequisite software

To be able to encrypt the data traffic between a z/OS LPAR and an accelerator, specific software components are required. On the z/OS side, various components of the IPsec module, which is available as part of the z/OS Communications Server, must be configured. The counterpart of the IPsec module on the accelerator consists of the libreswan module and a few updates to the Red Hat Linux operating system.

## Before you begin

To find the required versions of the software components that are discussed here, see the following website:

Prerequisites and Maintenance for IBM DB2 Analytics Accelerator for z/OS Version 5.1

## About this task

The accelerator components are delivered with the IBM PureData System for Analytics Host Management Kit. The upgrade of the Host Management Kit is carried out by an IBM service engineer. You must open a service request for that purpose.

## Procedure

1. Open a service request for an update of the Host Management Kit on your accelerator machines. The installation will be carried out with the help of a service engineer.
2. Make sure that the required IPsec components of the z/OS Communications Server are installed on the z/OS LPARs that are involved and that the necessary connections exist:
   - Each LPAR that is involved in the setup requires running instances of the following components:
     - Internet Key Exchange (IKE) daemon
     - Traffic Regulation Management (TRM) daemon
     - SYSLOG daemon
   - The LPAR that you use for managing certificates additionally requires a running instance of the Policy Agent and the Network Security Services daemon (NSSD).
   - An AT/TLS connection is required between the IKE daemon on the LPARs connecting to an accelerator and the Network Security Services (NSS) daemon on the LPAR that is used for managing the certificates.

     You find a sample configuration for two LPARs, an LPAR for encrypted connections to an accelerator and an LPAR for certificate management, in the appendix at the end of this document.

**Related reference**:

Appendix A, “Appendix: sample configuration of an AT/TLS connection,” on page 37

## How to avoid locking yourself out

You might lock yourself out of a z/OS LPAR or accelerator if you activate encryption-related configuration changes for a participating machine at the wrong time. Follow one of the procedures here to avoid this.

### Procedure

- If you have more than one logical partition (LPAR) connected to the same accelerator, and you want to enable encryption for the network communication between one of the LPARs and the accelerator, always initiate the configuration changes by connecting to the IBM DB2 Analytics Accelerator Console from an LPAR that is not involved in the process.
- If you have just one LPAR, follow this procedure:
  1. Prepare all configuration changes on your z/OS data server, but do not yet activate them.
  2. Connect to the accelerator and initiate the necessary configuration changes there. This activates a timer on the accelerator. The timer disables encryption after a certain period unless you stop the timer before this period has elapsed. This gives you a chance to reconnect to the machine and correct any configuration errors.
  3. Activate the configuration changes on your z/OS LPAR.
  4. Test the connection to the accelerator and verify all configuration changes.
  5. After a successful test, reconnect to the accelerator and confirm the changes, thereby stopping the timer.

## Generating a key pair and a certificate for a z/OS LPAR

You start by creating a pair of keys and a certificate to be used by each z/OS LPAR that is supposed to participate in the encrypted network communication. This certificate is stored in a key ring on the Network Security Services (NSS) server LPAR, that is, the LPAR on which the NSS daemon is running.

### Before you begin

Make sure that the required IPsec components of the z/OS Communications Server are installed and operational.

### About this task

This task is optional if you have already configured another IPsec connection from the same logical partition (LPAR).

### Procedure

1. Create an AT/TLS connection between the Internet Key Exchange (IKE) daemon that runs on the client LPAR that you want to enable encryption for, and the LPAR that hosts the NSS server. Verify this connection from the client LPAR that runs the IKE daemon by using the following command:

   ```
   ipsec -w display
   ```

   See the appendix of this document for a sample configuration of such a connection.

2. When the AT/TLS connection is ready, open a suitable editor to create a JCL job to be run on the NSS server LPAR (in the example, the name of this LPAR is ACQ1).
3. If a suitable NSS key ring does not yet exist, create one. Include, for example, the following code in the JCL job:

```
//* add a ring

RACDCERT ID(NSSD   ) ADDRING(IDAARING)
```

4. Add commands to the JCL that will create a self-signed root (CA) certificate, for example:

```
//* create a CA certificate (self-signed)

 RACDCERT CERTAUTH GENCERT                          -
   SUBJECTSDN(                                      -
     O('XYZ Corporation')                           -
     OU('SYSTEM Z TESTLAB')                         -
     C('US'))                                       -
   NOTBEFORE(DATE(2015-08-06))                      -
   NOTAFTER(DATE(2020-03-15))                       -
   KEYUSAGE(CERTSIGN)                               -
   WITHLABEL('IKED_CA1_IPSEC')
```

5. Add a command that will add the certificate to the key ring, for example:

```
//* add the CA certificate to the ring

RACDCERT ID(NSSD) CONNECT(CERTAUTH LABEL('IKED_CA1_IPSEC') +
         RING(IDAARING) USAGE(CERTAUTH))
```

6. Add commands to generate a pair of keys and a certificate that the IKE daemon on the client LPAR can use to authenticate itself to an accelerator, for example:

```
RACDCERT ID(NSSD) GENCERT                  -
    SUBJECTSDN(CN('DEAPQ1_IPSEC')         -
    OU('ZOS')                              -
    O('IBM')                               -
    L('BB') SP('BW') C('DE'))              -
    NOTAFTER(DATE(2020-03-15))             -
    SIZE(2048)      WITHLABEL('APQ1_IPSEC') -
    SIGNWITH(CERTAUTH LABEL('IKED_CA1_IPSEC'))

RACDCERT ID(NSSD) CONNECT(ID(NSSD) LABEL('APQ1_IPSEC')-
         RING(IDAARING))
```

7. Add commands to create a RACF profile with appropriate permissions, so that the IKE daemon can use the newly created pair of keys. The following sample job creates a profile for a user called APQ1IP1. That user ID is used by an IKE daemon on a client LPAR named APQ1. This client LPAR connects to the NSS server on the LPAR where the profile is stored (ACQ1). The daemon runs in a TCP/IP stack called TCPIP, and uses a key pair called APQ1_IPSEC, which is managed by the NSS server on this LPAR called ACQ1:

```
ADDUSER APQ1IP1 NAME('NSSD CLIENT')                      -
         UACC(NONE) DFLTGRP(USERS  )                     -
         DATA('NSSD CLIENT USED FOR IKED/IPSEC')         -
         OMVS(UID(20) HOME('/u/apq1ip1') PROGRAM('/bin/sh'))-
         OWNER(DEPT1234)

RDEF SERVAUTH EZB.NSS.ACQ*.APQ1_TCPIP.IPSEC.CERT    UAC(NONE)

RDEF SERVAUTH EZB.NSSCERT.ACQ1.APQ1_IPSEC.HOST    UACC(NONE)

PE EZB.NSS.ACQ*.APQ1_TCPIP.IPSEC.CERT    CL(SERVAUTH)-
     ID(ACP1IP1) ACC(READ)
```

```
PE EZB.NSSCERT.ACQ1.APQ1_IPSEC.HOST CL(SERVAUTH) -
     ID(APQ1IP1) ACC(READ)

SETROPTS RACLIST(SERVAUTH) REFRESH
SETROPTS GENERIC(SERVAUTH) REFRESH
```

**Note:** The same names are used in the `NssStackConfig` client stanza of the `iked.conf` file on the client LPAR APQ1. According to the previous example, the `Userid` in that file would be `APQ1IP1`, and the `ClientName` would be `APQ1_TCPIP`. That is, these are details to be used by the client LPAR.

8. You can optionally activate Remote Management services. When Remote Management is enabled, the IPsec commands are routed directly to the NSS server LPAR, which then acts on behalf of the NSS client (the IKE daemon on the client LPAR):

```
RDEF SERVAUTH EZB.NSS.ACQ*.APQ1_TCPIP.IPSEC.NETMGMT UAC(NONE)

PE EZB.NSS.ACQ*.APQ1_TCPIP.IPSEC.NETMGMT CL(SERVAUTH)-
     ID(APQ1IP1) ACC(READ)
```

9. Submit the JCL job.

**Related reference**:

Appendix A, “Appendix: sample configuration of an AT/TLS connection,” on page 37

## Generating a key pair and a certificate for an accelerator on the NSS server LPAR

A further pair of keys and certificate is required for the accelerators that are supposed to participate in the encrypted network communication with a z/OS client LPAR. An accelerator requires the RSA key pair and the associated certificate in a PKCS#12 password-encrypted file. You can use a tool of choice or an external certificate authority to generate the PKCS#12 file. This chapter contains instructions how to generate the PKCS#12 file using the z/OS Security Server RACF RACDCERT command.

### Before you begin

See RACDCERT GENCERT (Generate certificate) for information about the authorizations that are required to run the RADCERT command.

### About this task

This task is optional if you have already configured another IPsec connection to the same accelerator.

**Note:** The same certificate can be used for all connections to the same accelerator.

### Procedure

1. Create a JCL job and add commands to generate a pair of keys and a certificate to contain the public key for an accelerator, so that the z/OS client LPAR can identify the accelerator as an authorized participant in the encrypted network communication. For example:

```
//* generate a user certificate for accelerator Q100 signed by the CA

RACDCERT ID(NSSD) GENCERT                -
    SUBJECTSDN(CN('Q100')                -
    OU('IDAA')                           -
```

```
O('IBM')                                -
L('BB') SP('BW') C('DE'))               -
NOTAFTER(DATE(2020-03-15))              -
SIZE(2048)      WITHLABEL('Q100')       -
SIGNWITH(CERTAUTH LABEL('IKED_CA1_IPSEC'))
```

In this example, a key pair and a certificate with an alias name of Q100 (`WITHLABEL('Q100')`) are created. An alias (or X.509 friendly name) is required to refer to the certificate when you enable IPsec communication. The certifcate is signed by a certificate authority (CA) named IKED_CA1_IPSEC.

2. Add commands to the JCL that will store the key pair and the certifcate in a PKCS#12 file that can be transferred and read by the accelerator, for example:

```
//* export the key pair and certificate into a PKCS#12 file
//* that can be transferred to the accelerator

 racdcert export(label('Q100')) ID(NSSD)           +
 DSN('IDAA.CERTS.Q100') FORMAT(PKCS12DER) PASSWORD('PASSWORD')
```

**Important:**

- Keep the password of the PKCS#12 file secret. Everyone who has access to the file and the password can access the private key for the authentication of the IPsec connection and thus use the key to run an attack on the encrypted traffic.
- Only the following characters are allowed for the password of the PKCS#12 file:
  - a-z
  - A-Z
  - 0-9
  - Underscore ( _ )

3. Submit the job. The result is a sequential data set.
4. Open a Unix System Services (USS) shell and create a copy of the sequential data set (containing the certificate) in the USS file system of an LPAR that is connected to the accelerator, so that the certificate can be transferred. Place the copy in the directory that the AQT_HOST_PACKAGE_DIRECTORY environment variable points to. For example:

```
cp "//'IDAA.CERTS.Q100'" /SYSTEM/local/idaatest/softwareupdates/q100.p12
```

**Remember:** The AQT_HOST_PACKAGE_DIRECTORY environment variable is set in the AQTENV data set used by the Workload Manager (WLM) environment of the IBM DB2 Analytics Accelerator stored procedures.

## Transferring the certificate to an accelerator

Transfer the certificate in PKCS#12 format to a connected accelerator to enable the mutual authentication of the accelerator and the associated z/OS client LPAR.

### Before you begin

In the task described here, the SYSPROC.ACCEL_UPDATE_SOFTWARE stored procedure is called from IBM DB2 Analytics Accelerator Studio. Hence the user who connects from the studio to the relevant LPAR must have the privilege to run this stored procedure.

A PKCS#12 file, which contains an RSA 2048-bit private/public key pair and a certificate, chains up to the root certificate in the UNIX System Services (USS) file system of the LPAR that is connected to an accelerator. The AQT_HOST_PACKAGE_DIRECTORY environment variable (specified in the

AQTENV data set , which is used by the WLM environment of the IBM DB2 Analytics Accelerator stored procedures) must point to the USS directory in which the PKCS#12 file is located.

### About this task

This task is optional if you have already configured another IPsec connection to the same accelerator.

**Note:** The same certificate can be used for all connections to the same accelerator.

### Procedure

1. Start IBM DB2 Analytics Accelerator Studio.
2. Open the Accelerator view of the accelerator that you want to transfer the certificate to.
3. If necessary, expand the **About** section.
4. Click **Transfer updates**.
5. In the Transfer Updates window, select the check box in front of the certificate name.
6. Click **Transfer**.

### Results

The certificate file is transferred to the `/nz/dwa/transfer/accelerator` directory on the selected accelerator.

### What to do next

You must import the transferred file into the NSS database of the accelerator. You do this from the IBM DB2 Analytics Accelerator Console.

**Related tasks**:

"Generating a key pair and a certificate for an accelerator on the NSS server LPAR" on page 7

## Importing a certificate into the Netezza NSS database

A certificate must be stored in the Netezza Network Security Services (NSS) database before it can be used for authentication purposes.

### Before you begin

To proceed, you need the following passwords:

- Password of the certificate (PKCS#12) file
- Password of the IBM DB2 Analytics Accelerator Console

### About this task

This task is optional if you have already configured another IPsec connection to the same accelerator.

**Note:** The same certificate can be used for all connections to the same accelerator.

### Procedure

1. Start and log on to the IBM DB2 Analytics Accelerator Console. For more information, see *Logging on to the IBM DB2 Analytics Accelerator Console* in the IBM Knowledge Center.
2. Type the number in front of the option `Manage Encryption of Data in Motion` and press Enter to display the submenu:

```
 main -> <#>
 ------------------------------------------------------
You have the following options:
 (0) - Go back one level
 (1) - Import PKCS#12 file with RSA keys and certificates
 (2) - Delete certificates
 (3) - Enable encryption of data in motion
 (4) - Confirm enablement of encryption
 (5) - Disable encryption of data in motion
 (6) - Display the status of encrypted connections
 (7) - Restart the IPSec service
```

3. Type 1 and press Enter: You see a screen similar to the following:

```
Select the PKCS#12 file to be imported.
You can transfer additional files using the Software Update
'Transfer' dialog in DB2 Analytics Accelerator Studio.

Files available in directory '/nz/dwa/transfer/accelerator':

1: ibmhostmgmt.package.tar.z
2: q100.p12

Select a file or enter 0 to go back: (Default 0) >
```

4. Type the number in front of the PCKS#12 file name and press Enter. In the previous example, this is the number 2. You are asked to enter the password of the PKCS#12 file:

   ```
   Enter password for PKCS#12 file (in TSO, use PF3 to hide input):
   ```

5. Enter the password of the PKCS#12 file. A text similar to the following is displayed if the password was correct:

```
PKCS12 IMPORT SUCCESSFUL

The NSS database now contains the following certificates:

Certificate Name                                             Trust Attributes
                                                             SSL,S/MIME,JAR/XPI
SYSTEM Z SW TESTLAB - XYZ Corporation                        ,,
Q100                                                         u,u,u

Press <return> to continue
```

6. Press Enter to return to the previous menu.

## Restarting the IPsec service on an accelerator

Restart the IPsec service on an accelerator so that a newly imported certificate is used.

### Before you begin

This step requires a short outage of the accelerator in question, during which new network connections to the accelerator cannot be established. This means that users receive an error message when they try to submit new queries or load jobs.

However, the outage does not impact jobs that were already running before the restart. None the less, to avoid irritation, plan and communicate the outage ahead of time. You might want to stop the accelerator entirely before the restart by using the `-STOP ACCEL` command.

### Procedure

1. Start and log on to the IBM DB2 Analytics Accelerator Console.
2. Type the number in front of the option `Manage Encryption of Data in Motion` and press Enter to display the submenu:

```
 main -> <#>
 ------------------------------------------------------------
You have the following options:
 (0) - Go back one level
 (1) - Import PKCS#12 file with RSA keys and certificates
 (2) - Delete certificates
 (3) - Enable encryption of data in motion
 (4) - Confirm enablement of encryption
 (5) - Disable encryption of data in motion
 (6) - Display the status of encrypted connections
 (7) - Restart the IPSec service
```

3. Type 7 and press Enter: The following screen is displayed:

```
The IPSec service establishes encrypted network tunnels
between a z/OS LPAR and an accelerator.

You must restart the IPSec service to activate newly imported
certificates.

Mind that new connections for queries or table loads cannot be opened during a restart.
Users receive an error message when they try to do that.

Do you want to restart the service now (y/n)?
```

4. Type y and press Enter. The following text is displayed:

```
Restarting the IPSec service ...

The IPSec service has been restarted.

Press <return> to continue.
```

5. Press Enter to return to the previous menu.

## Configuring the Policy Agent on your client LPARs

The behavior of the IPsec service is controlled by a component that is called the Policy Agent. This agent executes a policy, which is a set of configurable instructions. You must adapt the policy for encrypted network traffic between a client LPAR and IBM DB2 Analytics Accelerator.

### Before you begin

Make sure that the following conditions apply:

- All IKE daemons must communicate with the NSS daemon on the NSS server LPAR over an AT/TLS connection.
- You have a list of all the IP addresses that must be configured:
  - A single accelerator has three IP addresses, a wall IP address, and one address each for the active and the inactive (standby) host.

  - Mostly, three IP addresses are used by a single z/OS client LPAR that is connected to an accelerator. These are the IP addresses of two OSA Express® cards, which are addressed through a third, common address. That common address is usually a static VIPA address.
  - A high-availability setup uses more than this basic set of IP addresses. Bear in mind that you must configure the Policy Agent for each IP address that is used.
  - If you use a high-availability setup for incremental updates or the *continuous incremental update* function, you must also configure the following addresses:
    - Dynamic (DVIPA) address that is used to connect to the LPARs that run instances of the CDC Capture Agent (address used for failover support)
    - Distributed (DDVIPA) addresses used for incremental-update-related traffic between the LPARs and the accelerator
- You know the distinguished names (DNs) of the X.500 certificates that are used to associate client LPARs and accelerators.

## About this task

The following steps show how to configure the Policy Agent for a single client LPAR and a single accelerator. It is a walk-through based on examples. The following IP addresses and DNs are used in the examples:

*Table 1. IP addresses and DNs used in examples*

| | |
|---|---|
| Name of accelerator | `Q100` |
| Wall IP address | `10.104.10.15` |
| Accelerator host 1 | `10.104.10.13` |
| Accelerator host 2 | `10.104.10.14` |
| DN of accelerator certificate | `CN=Q100,OU=IDAA,O=IBM, L=BB,ST=BW,C=DE` |
| VIPA address of LPAR | `10.104.11.5` |
| IP address of OSA Express card 1 | `10.104.11.3` |
| IP address of OSA Express card 2 | `10.104.11.4` |
| DN of certificate for IKE daemon | `CN=DEAQP1_IPSEC,OU=ZOS, O=IBM,L=BB,ST=BW,C=DE` |

## Procedure

1. Use TSO to log on to the z/OS client LPAR that connects to the accelerator.
2. Open the Policy Agent configuration-file in an editor, such as ISPF.
3. Add address sets consisting of the range of IP addresses for each accelerator. Each set must comprise the wall IP address and both host addresses. For example:

```
IpAddrSet Q100
{
   Range 10.104.10.13-10.104.10.15
}
```

4. Add the accelerator address sets to the list of IPsec address groups. For example:

```
IpAddrGroup swan-clients-IDAA
{
   IpAddrSetRef Q100
   IpAddrSetRef ...
}
```

5. Add an address group for the private data network on z/OS: For example:

```
IpAddrGroup mvsaddr-v4{
   IpAddr deaqp1_1
      {
      Addr 10.104.11.3
      }
   IpAddr deaqp1_2
      {
      Addr 10.104.11.4
      }
#  VIPA on deaqp1
   IpAddr deaqp1_vipa
      {
      Addr 10.104.11.5
      }
      }
```

6. Add a statement for security associations to be dynamically generated when data is routed through a virtual private network (VPN). This statement specifies how to encrypt the data. It also determines the mechanism to be used for authentication.

   **Important:** The following piece of code is not an example. You must specify it exactly as written because it matches the hard-wired IPsec configuration of the accelerator.

```
IpDynVpnAction SHA256-AES256-Transport
{
   VpnLife 0
   InitiateWithPfs    Group14
   AcceptablePfs      Group14
   IpDataOffer
   {
      HowToEncap              Transport
      HowToEncrypt            AES_GCM_16 KeyLength 128
      HowToAuth               ESP NULL
   }
   IpDataOffer
   {
      HowToEncap              Tunnel
      HowToEncrypt            AES_GCM_16 KeyLength 128
      HowToAuth               ESP NULL
   }
   IpDataOffer
   {
      HowToEncap              Transport
      HowToEncrypt            AES_GCM_16 KeyLength 256
      HowToAuth               ESP NULL
   }
   IpDataOffer
   {
      HowToEncap              Tunnel
      HowToEncrypt            AES_GCM_16 KeyLength 256
      HowToAuth               ESP NULL
   }
}
```

7. Add filter action statements which permit or deny certain types of network traffic and which also enable or disable logging for the various traffic types. For example:

```
IpGenericFilterAction PERMIT-NOLOG
{
   IpFilterAction PERMIT
   IpFilterLogging no
}


IpGenericFilterAction PERMIT-LOG
{
   IpFilterAction PERMIT
   IpFilterLogging yes
}

IpGenericFilterAction IPSEC-LOG-ALL
{
   IpFilterAction IPSEC
   IpFilterLogging yes
}
```

8. Add the following statements to the existing filter policy to include settings for the handling of security associations, the ports for traffic on the basis of the IKE protocol, the IPsec payload, and other data traffic. For example:

```
IpFilterPolicy
{
 ...
# allow dynamic activation of security associations with the accelerator
 AllowOnDemand          Yes

# allow IKE protocol to use UDB ports 4500 and 500
   IpFilterRule        ike-client-v4-permit
   {
     IpSourceAddrGroupRef mvsaddr-v4
     IpDestAddrGroupRef   swan-clients-IDAA
     IpService           IKE-NATT
     {
        Protocol             UDP
        SourcePortRange      4500
        Direction            Bidirectional
        Routing              LOCAL
     }
     IpService              IKE-NAPT
     {
        Protocol             UDP
        SourcePortRange      500
        Direction            Bidirectional
        Routing              LOCAL
     }
     IpGenericFilterActionRef   PERMIT-LOG
   }

# policy for IPsec payload traffic
   IpFilterRule  client-v4-ipsec-IDAA
   {
      IpSourceAddrGroupRef mvsaddr-v4
      IpDestAddrGroupRef swan-clients-IDAA
      IpService
      {
         Protocol       all
         Direction      bidirectional
         Routing        local
      }
      IpGenericFilterActionRef IPSEC-LOG-ALL
      IpDynVpnActionRef SHA256-AES256-Transport
   }
...
# do not forget a permit a rule at the end of your
IpFilterPolicy that allows all other traffic
```

```
   IpFilterRule                 Permitall
   {
      IpSourceAddr              All
      IpDestAddr                All
      IpService
      {
         Protocol               All
         Direction              bidirectional
         Routing                Either
      }
      IpGenericFilterActionRef  PERMIT-NOLOG
   }
 ...
}
```

9. Add a key exchange action, which specifies which protocol and mechanism to use for the authentication of the IKE daemons. For example:

```
KeyExchangeAction AES-SHA2-DH14-RSA
{
   HowToInitiate      IKEv2
   HowToRespond       Either
   HowToAuthMe        RsaSignature
   KeyExchangeOffer
   {
      HowToEncrypt            AES_CBC KeyLength 256
      HowToAuthMsgs           SHA2_256
      PseudoRandomFunction    HMAC_SHA2_256
      HowToVerifyMsgs         HMAC_SHA2_256_128
      HowToAuthPeers          RsaSignature
      DHGroup                 Group14
   }
}
```

10. Add a *local security endpoint*, which specifies the certificate that the IKE daemon must use to authenticate the IP addresses of the client LPAR. This is done by referencing the name of the certificate that is managed by the NSS. For example:

```
 LocalSecurityEndpoint  mvs-deaqp1-rsa
      {
         Identity X500dn CN=BOEDWB1_IPSEC,OU=ZOS,O=IBM,L=BB,ST=BW,C=DE
         LocationGroupRef     mvsaddr-v4
      }
```

11. Similarly, add a *remote security endpoint*, which specifies the certificate to be used for the authentication of the accelerator IP addresses. For example:

```
RemoteSecurityEndpoint rse_Q100
{
        Identity X500dn CN=Q100,OU=IDAA,O=IBM,L=BB,ST=BW,C=DE
        LocationSetRef Q100
}
```

    **Tip:** If you have more than one certificate for different accelerators, you can use a wildcard (*) in the place of the certificate name to define a common remote security endpoint for all accelerators. For example:

```
RemoteSecurityEndpoint rse_idaa_swan_clients
{
        Identity X500dn CN=DEAQP1_IPSEC,OU=ZOS,O=IBM,L=BB,ST=BW,C=DE
        LocationGroupRef swan-clients-IDAA
}
```

    Make sure to use the common endpoint name in the key exchange rule that you set in the next step.

12. Add a rule to the existing key exchange policy that references the newly created local security endpoint and remote security endpoint. For example:

```
KeyExchangePolicy
{
...
      KeyExchangeRule swan-client-v4-q100-rsa
   {
      LocalSecurityEndpointRef mvs-deaqp1-rsa
      RemoteSecurityEndpointRef rse_Q100
      KeyExchangeActionRef AES-SHA2-DH14-RSA
   }

...
}
```

# Enabling encrypted connections on an accelerator

To enable encrypted network traffic between a z/OS client LPAR and an accelerator, you must, on the accelerator, associate that LPAR with an imported certificate. You do this on the IBM DB2 Analytics Accelerator Console.

## Before you begin

Make sure that the following steps have already been completed:

- You have transferred and imported a PKCS#12 certificate file for the authentication of the new connection.
- You have prepared the Policy Agent configuration for IPsec on the client LPARs that connect to the accelerator.

This step requires a short outage of the accelerator in question, during which new network connections to the accelerator cannot be established. This means that users receive an error message when they try to submit new queries or load jobs. However, the outage does not impact jobs that were already running before the restart. None the less, to avoid irritation, plan and communicate the outage ahead of time. You might want to stop the accelerator entirely before the restart by using the `-STOP ACCEL` command.

## About this task

Remember that each IP address in your setup must be configured. This also applies to the enablement on the accelerator side. Hence the following steps must be completed for each of these IP addresses:

- IP addresses of the OSA Express cards
- Common address for communication with the OSA Express cards (usually a static VIPA address)
- If you use a high-availability setup for incremental updates or the *continuous incremental update* function, you must also configure the following addresses:
  - Dynamic (DVIPA) address that is used to connect to the LPARs that run instances of the CDC Capture Agent (address used for failover support)
  - Distributed (DDVIPA) addresses used for incremental-update-related traffic between the LPARs and the accelerator

**Remember:** A high-availability setup uses more than this basic set of IP addresses. Bear in mind that you must also configure the Policy Agent for each IP address that is used.

## Procedure

1. Start and log on to the IBM DB2 Analytics Accelerator Console. For more information, see *Logging on to the IBM DB2 Analytics Accelerator Console* in the IBM Knowledge Center.
2. Type the number in front of the option `Manage Encryption of Data in Motion` and press Enter to display the submenu:

```
 main -> <#>
 ------------------------------------------------------
You have the following options:
 (0) - Go back one level
 (1) - Import PKCS#12 file with RSA keys and certificates
 (2) - Delete certificates
 (3) - Enable encryption of data in motion
 (4) - Confirm enablement of encryption
 (5) - Disable encryption of data in motion
 (6) - Display the status of encrypted connections
 (7) - Restart the IPSec service
```

3. Type 3 and press Enter: The following screen is displayed:

```
Important: You must enable encryption for each
TCP/IP address that is used by a z/OS LPAR to connect to an accelerator.

Before you continue, make sure that the following steps have already been completed:
- You have transferred and imported a PKCS#12 certificate file for
  the authentication of the new connection.
- You have configured the Policy Agent for IPsec on the z/OS side.

Specify a name for this configuration (allowed characters: A-Z, a-z, 0-9;
max. length 20). You might want to enter the LPAR name (or 0 to exit):
```

4. Type the name and press Enter. The following text is displayed:

```
Specify a z/OS IP address that the LPAR uses to connect to the
accelerator over the (private) data network. This is usually a
static VIPA address. (To exit, enter 0.):
```

5. Type the IP address and press Enter. For example:

   `10.104.11.5`

   You might see a text message like this:

   ```
   Address 10.104.11.5 cannot be reached from the accelerator.
   Do you want to continue anyway (y/n)? [n]:
   ```

6. Type y and press Enter. The following text is displayed:

```
Specify the name of the certificate that you
want to use to authenticate the accelerator for this
connection.('0' to exit):

Certificates available:
 1 : IDAASIM37

Select a certificate or enter 0 to go back:  (Default 0) >
```

7. To select the (only) certificate in this example, type 1 and press Enter. The following text is displayed:

```
This configuration will be disabled again unless you
confirm it during a timeout period.

Set the length of the timeout period.
Enter a value between 5 and 1440 minutes.
Press <return> to accept the default of 30 minutes.
Cancel the process by entering 0.
```

8. Type the number of minutes and press Enter. The following text is displayed:
   ```
   Enable encryption of data in motion now (y/n) [n]:
   ```
9. Type y and press Enter. The following text is displayed:
   ```
   The configuration is now active.

   Your next steps:

   - Activate your changed configuration of the IPsec service
     on the z/OS side.
   - Test the connection.
   - Re-connect to this console and confirm the connection.

   Done
   Press <return> to continue
   ```
10. Press Enter to return to the previous menu.
11. Repeat steps 3 on page 17 through 10 for the remaining IP addresses.

## Activating the changed configuration of the Policy Agent

To activate your changed Policy Agent configuration on a client LPAR connecting to an accelerator, you must restart or refresh the Policy Agent.

### Before you begin

Make sure that you have enabled the encryption of network traffic between the client LPARs and the connected accelerators (previous step, to be carried out on the accelerators).

### About this task

The NSS daemon does not necessarily run on the LPAR that you want to enable encryption for. So make sure that you log on to and submit the following commands for or from the right LPAR.

### Procedure

1. Use TSO to log on to the z/OS LPAR on which the NSS daemon is running.
2. From TSO or the SDSF command interface, submit a command to refresh the configuration of the IPsec Policy Agent. For example, the following command refreshes the IPsec configuration for an LPAR called APQ1:
   ```
   MODIFY NSSD,REFRESH,FILE=//'SYS1.APQ1.TCPIP.NSSD(CONF)'
   ```
3. Use TSO to log on to the z/OS client LPAR that connects to the accelerator.
4. From TSO or the SDSF command interface, submit the command to refresh the configuration of the IPsec Policy Agent:
   ```
   F PAGENT,REFRESH
   ```
5. If this has not been done yet, also make the NSS daemon re-read the NSS key ring (file), which contains the certificate that you want to use for the

authentication of the encryption-enabled LPAR. To this end, you must refresh the SERVAUTH service class from the LPAR on which the NSS daemon is running. For example:

```
SETROPTS RACLIST(SERVAUTH) REFRESH
```

## Testing the connection

After enabling an LPAR for encrypted network traffic on the accelerator side and activating the changed IPsec policy on the z/OS side, check whether you can still reach the machines that are supposed to take part in the encrypted network communication. Complete this step before you confirm the connection.

### Procedure

1. Use TSO to log on to the z/OS client LPAR that uses the newly configured IPsec connection to connect to an accelerator.
2. Open a UNIX System Services shell.
3. From the command line, ping the wall IP address of the accelerator, for example:

   ```
   ping 10.104.10.12
   ```

   If the ping was successful, a response similar to this one is displayed:

   ```
   CS V1R13: Pinging host 10.104.10.13
   Ping #1 response took 0.003
   ```

4. Repeat step 3 for each IP address that you enabled on the acceleator.
5. From TSO or the SDSF command interface in TSO, check whether you can still connect to the IBM DB2 Analytics Accelerator Console:

   ```
   TSO TELNET 10.104.10.13   1600
   ```

### What to do next

If all operations were successful, confirm the encrypted connection on the IBM DB2 Analytics Accelerator Console.

If one of the operations fails, click the appropriate link at the end of this topic for troubleshooting information.

**Remember:**

The accelerator disables the encrypted connection to the selected LPAR at the end of the timeout period. This allows you to connect again. You might have to undo your changes to the Policy Agent configuration before you reconnect to the accelerator.

**Related information**:

"The AT/TLS connection does not work" on page 34

"An IPsec tunnel cannot be established" on page 35

## Confirming an encrypted connection

Confirm a successfully tested encrypted connection on the IBM DB2 Analytics Accelerator Console before the end of the timeout period.

### Before you begin

Make sure that the following tests have been carried out successfully:

- You could ping the accelerator by using the wall IP address.
- You could connect to the IBM DB2 Analytics Accelerator Console from the client LPAR that you configured for encrypted network traffic to an accelerator.

**Attention:** Do not confirm encryption settings for a connection that you have not tested. Only confirm a connection if you are sure that it works. Otherwise, you might permanently lose access to the accelerator.

### Procedure

1. Start and log on to the IBM DB2 Analytics Accelerator Console. For more information, see *Logging on to the IBM DB2 Analytics Accelerator Console* in the IBM Knowledge Center.
2. Type the number in front of the option `Manage Encryption of Data in Motion` and press Enter to display the submenu:

```
 main -> <#>
 ------------------------------------------------------------
You have the following options:
 (0) - Go back one level
 (1) - Import PKCS#12 file with RSA keys and certificates
 (2) - Delete certificates
 (3) - Enable encryption of data in motion
 (4) - Confirm enablement of encryption
 (5) - Disable encryption of data in motion
 (6) - Display the status of encrypted connections
 (7) - Restart the IPSec service
```

3. Type `4` and press Enter. A screen similar to the following is displayed:

```
The following connections have been configured, but not yet confirmed:

Connection name     | z/OS IP address          | Timer expires in
--------------------+--------------------------+-----------------------
AQP1VIPA            | 10.101.14.5              |   27 minutes
AQP1VIPA            | 10.101.14.25             |   23 minutes

Specify the connection that you want to confirm by entering its name.
(Press Enter to go back without specifying a value.
Enter the value ALL to confirm all of the above connections):
```

4. Type the appropriate value and press Enter or just press Enter (no value) to cancel the process and return to the menu. If you specify a valid value, the affected connections are confirmed:

```
The timer for connection AQP1VIPA is now stopped and the configuration is
saved.
The timer for connection AQP1VIPA is now stopped and the configuration is
saved.
```

   In this example, two connections to the same LPAR, each of which using a different IP address, have been confirmed.
5. Press Enter to return to the previous menu.

## Verifying the encryption status

Having confirmed and thus saved the configuration for one or more connections, verify that the systems involved do in fact list these as encrypted connections..

## Procedure

- To verify the encryption status of your connections from z/OS, follow these steps:
  1. Use TSO to log on to a z/OS client LPAR that uses an encrypted IPsec connection to connect to an accelerator.
  2. You can use the `ipsec` command to list the encrypted connections (tunnels) or the IKE security associations. See the following examples:
     - `ipsec -p TCPIP -y display`

       This command lists the encrypted connections for a TCP/IP stack named TCPIP.
     - `ipsec -p TCPIP -k display`

       This command lists the IKE security associations for a TCP/IP stack named TCPIP.
- To verify the encryption status of your connections from an accelerator, follow these steps:
  1. Start and log on to the IBM DB2 Analytics Accelerator Console.
  2. Type the number in front of the option `Manage Encryption of Data in Motion` and press Enter to display the submenu:

```
 main -> <#>
 ------------------------------------------------------
You have the following options:
 (0) - Go back one level
 (1) - Import PKCS#12 file with RSA keys and certificates
 (2) - Delete certificates
 (3) - Enable encryption of data in motion
 (4) - Confirm enablement of encryption
 (5) - Disable encryption of data in motion
 (6) - Display the status of encrypted connections
 (7) - Restart the IPSec service
```

  3. Type `6` and press Enter: All IP addresses that are involved in your encrypted connections are listed once in each of the categories `ISAKMP`, `IPsec`, and `Encrypted traffic`. For example:

```
ISAKMP (Phase 1) Security Associations:
---------------------------------------
#5: "deaqp1_10.104.11.25_10.104.10.15" REPLACE in 19544s
#7: "deaqq1_10.104.11.3_10.104.10.15" REPLACE in 19586s

IPsec (Phase 2) Security Associations:
--------------------------------------
#6: "deaqp1_10.104.11.25_10.104.10.15" REPLACE in 5385s isakmp#5
#8: "deaqq1_10.104.11.3_10.104.10.15" REPLACE in 5177s isakmp#7

Encrypted traffic (Encapsulated Security Payload - ESP):
--------------------------------------------------------
#6: "deaqp1_10.104.11.25_10.104.10.15" in=119MB out=3937MB
#8: "deaqq1_10.104.11.3_10.104.10.15" in=28KB out=7KB


Press <return> to continue
```

- You can also view the encryption status from IBM DB2 Analytics Accelerator Studio:
  1. Open the Accelerator view of the accelerator that is supposed to use encrypted connections.
  2. In the header of the Accelerator view, click the **Encryption details** link. A window with the title Encryption Details for Accelerator <name> opens. It is devided into the following sections:

**Disk Encryption**
Shows whether disk encryption is used by your accelerator hardware.

**Network Encryption**
Shows two tables and a text box:

**Encrypted connections**
This table contains details about the encrypted connections (IPsec tunnels) to the selected accelerator. This includes the names of the connecting client LPARs, their IP addresses, the IP addresses of the active and the passive accelerator host, the names of the certificates that are used, and the date and time when the connections were established.

**Certificates**
This table shows details about the certificates that are used, such as the validity, the type (user or root certificate), and the distinguished name.

**Selected certificate in X.509 encoding**
Shows the content of a certificate that is selected in the table above, in (encrypted) X.509 Base64 encoding. Naturally, the displayed content does not include the private keys because these must remain secret.

3. If a connection does not work, the **Accelerator IP address** column in the **Encrypted connections** table shows *no active tunnel*.

# Disabling encryption of data in motion

In case that you no longer want to encrypt network traffic between your z/OS client LPARs and your accelerators, you can, of course, undo the configuration changes and disable encryption of data in motion. In doing so, it is crucial that you observe the recommended order of steps closely to avoid losing access to a system.

## Removing accelerator-related entries from the Policy Agent configuration file

Remove entries from your Policy Agent configuration-file as shown, but do not activate the modified policy before you have also changed the IPsec configuration on the connected accelerators.

### Procedure

1. Use TSO to log on to the z/OS client LPAR that you want to disable.
2. Open the Policy Agent configuration-file in an editor, such as ISPF.
3. Remove the following entries from the configuration file of the Policy Agent, but only if these entries pertain to the very last accelerator that you are going to remove. If you still want to use encrypted connections for a subset of your accelerators, skip this step and continue with step 4 on page 24. For example:

```
# allow IKE protocol to use UDB ports 4500 and 500
   IpFilterRule        ike-client-v4-permit
   {
     IpSourceAddrGroupRef mvsaddr-v4
     IpDestAddrGroupRef   swan-clients-IDAA
     IpService           IKE-NATT
     {
        Protocol             UDP
        SourcePortRange      4500
        Direction            Bidirectional
        Routing              LOCAL
     }
     IpService              IKE-NAPT
     {
        Protocol             UDP
        SourcePortRange      500
        Direction            Bidirectional
        Routing              LOCAL
     }
     IpGenericFilterActionRef   PERMIT-LOG
   }

# policy for IPsec payload traffic
   IpFilterRule  client-v4-ipsec-IDAA
   {
      IpSourceAddrGroupRef mvsaddr-v4
      IpDestAddrGroupRef swan-clients-IDAA
      IpService
      {
         Protocol       all
         Direction      bidirectional
         Routing        local
      }
      IpGenericFilterActionRef IPSEC-LOG-ALL
      IpDynVpnActionRef SHA256-AES256-Transport
   }
```

```
The LocalSecurityEndpoint referencing the z/OS certificate LocalSecurityEndpoint
    mvs-boedwb-rsa
      {
         Identity X500dn CN=BOEDWB1_IPSEC,OU=ZOS,O=IBM,L=BB,ST=BW,C=DE
         LocationGroupRef     mvsaddr-v4
      }
```

4. For the accelerators that you want to disable, remove the IP address sets (`IpAddrSet`): For example, remove:

```
IpAddrSet Q100
{
   Range 10.104.10.13-10.104.10.15
}
```

5. Also remove the references in the `IpAddrGroup` section that list the IPsec peers: For example:

```
IpAddrGroup swan-clients-IDAA
{
   IpAddrSetRef Q100
   IpAddrSetRef ...
}
```

6. For the accelerators that you want to disable, remove the `KeyExchangeRule` entries in the `KeyExchangePolicy` section: For example, remove:

```
KeyExchangePolicy
{
...
   KeyExchangeRule swan-client-v4-q100wallip-rsa
   {
      LocalSecurityEndpointRef mvs-boedwb-rsa
      RemoteSecurityEndpoint q100wallip
      {
        Identity X500dn CN=Q100,OU=IDAA,O=IBM,L=BB,ST=BW,C=DE
        Location 10.101.13.15
      }
      KeyExchangeActionRef AES-SHA2-DH14-RSA
   }

   KeyExchangeRule swan-client-v4-q100host1-rsa
   {
      LocalSecurityEndpointRef mvs-boedwb-rsa
      RemoteSecurityEndpoint q100host1
      {
        Identity X500dn CN=Q100,OU=IDAA,O=IBM,L=BB,ST=BW,C=DE
        Location 10.101.13.13
      }
      KeyExchangeActionRef AES-SHA2-DH14-RSA
   }
   KeyExchangeRule swan-client-v4-q100host2-rsa
   {
      LocalSecurityEndpointRef mvs-boedwb-rsa
      RemoteSecurityEndpoint q100host2
      {
        Identity X500dn CN=Q100,OU=IDAA,O=IBM,L=BB,ST=BW,C=DE
        Location 10.101.13.14
      }
      KeyExchangeActionRef AES-SHA2-DH14-RSA
   }
...
}
```

## Alternative: Switching to the default TCP/IP filter rules

Instead of removing entries from the Policy Agent configuration-file, you can switch to the default filter rules that are defined in the TCP/IP profile (usually a data set called TCPIP.PROFILE). Use this method as a fallback solution.

### Before you begin

- Use this approach only if no other connection to the same LPAR uses IPsec encryption.
- Make sure that the TCP/IP profile contains appropriate IPsec rules (keyword `IPSECRULES`):

### Procedure

- Turn on default TCP/IP filter rules by using the **ipsec** command:

  ```
  ipsec -f default   # TCPIP profile filters
  ```

  If, for some reason, you must revert to IPsec filtering, enter:

  ```
  ipsec -f reload    # IPsec policy filters
  ```

- Another alternative is to use a second Policy Agent configuration file and to refer to that file in the `IpSecConfig` statement of the `TCPIP.image` file. Suppose, for example, that you have two Policy Agent configuration files. The first first has been stripped of all references to your "encrypted accelerators", while the second, for backup purposes, still contains these entries:
  - `/path/ipsec_without_accelerators.policy`
  - `/path/ipsec_with_accelerators.policy`

  To refer to the first of these policy files in `TCPIP.image`, the `TCPIP.image` file must include the following line:

  ```
  IpSecConfig /path/ipsec_without_accelerators.policy
  ```

  In addition, your `/etc/pagent.conf` file must contain an entry that refers to the `TCPIP.image` file, for example:

  ```
  TcpImage TCPIP /etc/policy/TCPIP.image
  ```

  To activate changes made in this way, submit the following command from TSO or the SFDF command interface:

  ```
  F PAGENT,REFRESH
  ```

## Disabling encrypted connections to a z/OS client LPAR on an accelerator

To disable encrypted network traffic between a z/OS client LPAR and an accelerator, remove the IPsec encryption settings for this LPAR from the accelerator.

### About this task

This step requires a short outage of the accelerator in question, during which new network connections to the accelerator cannot be established. This means that users receive an error message when they try to submit new queries or load jobs. However, the outage does not impact jobs that were already running before the restart. None the less, to avoid irritation, plan and communicate the outage ahead of time. You might want to stop the accelerator entirely before the restart by using the `-STOP ACCEL` command.

### Procedure

1. Start and log on to the IBM DB2 Analytics Accelerator Console. For more information, see *Logging on to the IBM DB2 Analytics Accelerator Console* in the IBM Knowledge Center.
2. Type the number in front of the option `Manage Encryption of Data in Motion` and press Enter to display the submenu:

```
main -> <#>
 ------------------------------------------------------
You have the following options:
 (0) - Go back one level
 (1) - Import PKCS#12 file with RSA keys and certificates
 (2) - Delete certificates
 (3) - Enable encryption of data in motion
 (4) - Confirm enablement of encryption
 (5) - Disable encryption of data in motion
 (6) - Display the status of encrypted connections
 (7) - Restart the IPSec service
```

3. Type 5 and press Enter: You see a screen similar to this here:

```
The following IPsec connections are active on this accelerator:

Connection name     | z/OS IP address          | Certificate
--------------------+--------------------------+-----------------------
1: AQP1VIPA         | 10.104.11.5              | Q100
2: AQQ1VIPA         | 10.104.11.25             | Q100

Specify the connection name that you want to delete
(to go back, press Enter without specifying a value): AQP1VIPA
```

4. Type the number in front of the connection that you want to disable, for example 1, and press Enter. The following text is displayed:

```
The following IPsec connections are now active on this accelerator:

Connection name     | z/OS IP address          | Certificate
--------------------+--------------------------+-----------------------
1: AQQ1VIPA         | 10.104.11.25             | Q100

Press <return> to continue
```

5. Press Enter to return to the previous menu.

### What to do next

To remove another encrypted connection, repeat steps 3 through 5.

## Activating the trimmed-down IPsec configuration

After the removal of IPsec encryption settings from one or more accelerators, activate the changed Policy Agent configuration on the client LPAR to make sure that it still works for other applications.

### Before you begin

Make sure that you have disabled encrypted connections to the relevant client LPAR on all connected accelerators.

### Procedure

1. Use TSO to log on to the z/OS client LPAR that connects to the accelerator.
2. From TSO or the SDSF command interface, submit the command to refresh the configuration of the IPsec Policy Agent:

   ```
   F PAGENT,REFRESH
   ```

### Results

The Policy Agent picks up the current policy configuration and data encryption for the remaining applications remains intact.

**Hint:** You need not refresh the SERVAUTH class to re-read the key-ring file at this point because the certificate for the accelerator is still in place. So there would be no change.

## Verifying the removal of encrypted accelerator connections

Having removed the encryption settings for IBM DB2 Analytics Accelerator and activated the trimmed-down IPsec service, verify that the systems involved no longer list enctypted connections to an accelerator.

### Procedure

- To verify the encryption status of your connections from z/OS, follow these steps:
  1. Use TSO to log on to the z/OS client LPAR that connects to the accelerator.
  2. You can use the `ipsec` command to list the encrypted connections (tunnels) or the IKE security associations. See the following examples:
     - `ipsec -p TCPIP -y display`

       This command lists the encrypted connections for a TCP/IP stack named TCPIP.
     - `ipsec -p TCPIP -k display`

       This command lists the IKE security associations for a TCP/IP stack named TCPIP.
  3. Make sure that connections to accelerators are not listed anymore.
- To verify the encryption status of your connections from an accelerator, follow these steps:
  1. Start and log on to the IBM DB2 Analytics Accelerator Console.
  2. Type the number in front of the option `Manage Encryption of Data in Motion` and press Enter to display the submenu:

```
main -> <#>
------------------------------------------------------------
You have the following options:
 (0) - Go back one level
 (1) - Import PKCS#12 file with RSA keys and certificates
 (2) - Delete certificates
 (3) - Enable encryption of data in motion
 (4) - Confirm enablement of encryption
 (5) - Disable encryption of data in motion
 (6) - Display the status of encrypted connections
 (7) - Restart the IPSec service
```

  3. Type `6` and press Enter: None of the formerly configured IP addresses must be shown anymore in the ISAKMP, IPsec, and Encrypted traffic categories.

## Deleting an accelerator certificate

It is recommended that you delete certificates from the Netezza NSS database on an accelerator if these certificates are no longer used.

### Before you begin

You need the password of the IBM DB2 Analytics Accelerator Console.

### Procedure

1. Start and log on to the IBM DB2 Analytics Accelerator Console. For more information, see *Logging on to the IBM DB2 Analytics Accelerator Console* in the IBM Knowledge Center.
2. Type the number in front of the option `Manage Encryption of Data in Motion` and press Enter to display the submenu:

```
 main -> <#>
 ------------------------------------------------------
You have the following options:
 (0) - Go back one level
 (1) - Import PKCS#12 file with RSA keys and certificates
 (2) - Delete certificates
 (3) - Enable encryption of data in motion
 (4) - Confirm enablement of encryption
 (5) - Disable encryption of data in motion
 (6) - Display the status of encrypted connections
 (7) - Restart the IPSec service
```

3. Type 2 and press Enter: You see a screen similar to the following:

```
The NSS database contains the following certificates:

Certificate Name                                              Trust Attri
                                                              SSL,S/MIME,

SYSTEM Z SW TESTLAB - IBM DEUTSCHLAND RESEARCH & DEVELOPMENT GMBH C,C,C
Q100                                                          u,u,u


Specify the name of the certificate to be deleted.
Do NOT delete a certificate that is still referenced by an encrypted
connection (to go back, press Enter without specifying a value):
```

4. Type the name of the certificate that you want to delete and press Enter. In the previous example, the name of the certificate is `Q100`. The text that is displayed does not list the deleted certificate anymore, as in the example:

```
The NSS database now contains the following certificates:

Certificate Name                                              Trust Attri
                                                              SSL,S/MIME,

SYSTEM Z SW TESTLAB - IBM DEUTSCHLAND RESEARCH & DEVELOPMENT GMBH C,C,C

Press <Return> to continue
```

5. Press Enter to return to the previous menu.

## Deleting a certificate from an NSS key ring

If you deleted a certificate from an accelerator, also delete its counterpart from the NSS key ring that is stored on the NSS server LPAR.

### Procedure

1. Use TSO to log on to the LPAR on which the NSS daemon is running.
2. Create a JCL job and add a command that will delete the certificate and the associated pair of keys from the key ring, for example:

```
RACDCERT  ID(NSSD) DELETE(LABEL('IKED_CA1_IPSEC'))
```

3. Submit the JCL job.
4. After you have removed the last accelerator from the Policy Agent configuration of all LPARs that use this certificate, you can also delete the associated RACF profiles. For example:

   ```
   RDEL SERVAUTH EZB.NSS.ACQ*.APQ1_TCPIP.IPSEC.CERT
   RDEL SERVAUTH EZB.NSSCERT.ACQ1.APQ1_IPSEC.HOST
   ```

   **Note:** The same names are used in the `NssStackConfig` client stanza of the `iked.conf` file on the LPAR ACP1. According to the previous example, the `Userid` in that file would be `APQ1IP1`, and the `ClientName` would be `APQ1_TCPIP`.
5. If the IKE connection to the NSS server is no longer needed, you can also delete the user (here: APQ1IP1) that you created for the IKE client, including its data-set profile and alias entry.

# Replacing an expiring certificate

An accelerator issues a warning when an encryption certificate is about to expire. Replace an expiring certificate before the passing of the expiration date because you will be unable to access the system afterwards.

# Troubleshooting

The following sections provide instructions on how to enable tracing for encrypted connections and how to obtain diagnostic information about particular problems.

## Tracing encrypted accelerator connections

The trace function in IBM DB2 Analytics Accelerator Studio now offers an option that allows you to collect encryption-related trace information about an accelerator. This information might prove useful in diagnosing problems with encrypted connections to and from an accelerator.

### Before you begin

Make sure that you use the correct version of IBM DB2 Analytics Accelerator Studio, that is, a version that provides the additional trace option for encryption. For more information, see:

Prerequisites and Maintenance for IBM DB2 Analytics Accelerator for z/OS Version 5.1

### About this task

If you cannot reach an accelerator because the encrypted connection does not work or you have locked yourself out accidentally, try to connect from IBM DB2 Analytics Accelerator Studio via a different LPAR that can still communicate with the accelerator. If no such LPAR is available, wait until the timeout period has run out and encryption is disabled again. This presupposes that you have not yet confirmed the encryption settings on that accelerator.

### Procedure

1. Start IBM DB2 Analytics Accelerator Studio.
2. Open the Accelerator view of the accelerator that you want to collect trace information about.
3. In the header of the Accelerator view, click **Save**.
4. In the Save Trace window, select **Network encryption (IPsec)**.
5. Click **Finish**.

### Results

If the **Network encryption (IPsec)** option is seleted, the compressed trace archive that is produced by IBM DB2 Analytics Accelerator Studio contains an additional directory named `IPsec` with the following files pertaining to the active accelerator host:

**`version.txt`**
Contains the version of the IBM PureData System for Analytics Host Management component

**`rpm-q.txt`**
A list of the installed IPsec components.

If version 5.1.4 or higher of the IBM PureData System for Analytics (PDA) Host Management and the libreswan component are installed, you find the following additional files in the IPsec directory of the trace archive:

**ipsec-barf.txt**
A lot of diagnostic information that is collected by running the **ipsec barf** command on a UNIX system. For more information, follow the link at the end of this topic.

**certutil-L.txt**
A list of the certificates in the NSS database of the accelerator

The IPsec directory in the trace archive contains the same set of files for the passive accelerator host, that is:
- version-passive-host.txt
- rpm-q-passive-host.txt
- ipsec-barf-passive-host.txt (PDA Host Management 5.1.4 or higher plus libreswan required )
- certutil-L-passive-host.txt (PDA Host Management 5.1.4 or higher plus libreswan required )

**Related information**:

https://libreswan.org/man/barf.8.html

# The AT/TLS connection does not work

The AT/TLS connection between the IKE daemon and the NSS daemon does not work.

## Symptoms

The certificate in the NSS key ring cannot be accessed, and therefore participating accelerators and LPARs cannot authenticate each other. As a result, encrypted connections cannot be established.

## Diagnosing the problem

To check the AT/TLS connection, run the following command on the z/OS LPAR that you use for managing the certificates (the LPAR running the NSS daemon). Use a USS shell or terminal connection to enter the command:

```
ipsec -w display
```

The screen output of this command is similar to this:

```
CS V2R1 ipsec  NSS Client Name: n/a  Tue Sep  8 17:59:02 2015
Primary:  Stack NSS       Function: Display           Format:   Detail
Source:   IKED            Scope:    n/a               TotAvail: 1
SystemName: DWP1

StackName:                      TCPIP
ClientName:                     DWP1_TCPIP
ClientAPIVersion:               4
ServerAPIVersion:               4
NSServicesSupported:            Yes
RemoteManagementSelected:       Yes
RemoteManagementEnabled:        Yes
CertificateServicesSelected:    Yes
CertificateServicesEnabled:     Yes
NSClientIPAddress:              9.152.87.129
NSClientPort:                   5203
NSServerIPAddress:              9.152.87.129
NSServerPort:                   4159
```

```
NSServerSystemName:              DWP1
UserID:                          DWP1IP1
ConnectionState:                 connected
TimeConnectedToNSServer:         2015/09/04 16:34:37
TimeOfLastMessageToNSServer:     2015/09/08 17:36:08
************************************************************************
```

If the connection works, the `ConnectionState` shows the state `connected`. If the status is different check the following log files for error messages:

- `SYSLOG` and `iked.log` on the client LPAR
- `nssd.log` on the NSS server LPAR

If these log files do not give you sufficient information, increase the value of `IkeSysLogLevel` to `255` in the configuration file of the IKE daemon on the client LPAR (file name: `iked.conf`) . This setting gives you more detailed log messages. Save the file and refresh the configuration of the IKE daemon to let the changes take effect.

### Resolving the problem

1. Try to solve the problem by using the information in the log (error) messages.
2. If you cannot find the cause of the error using the log messages, verify that the configuration steps in Steps for authorizing resources for NSS were followed correctly.
3. Compare your setup with the sample AT/TLS configuration that is provided in this document.
4. If this does not help, contact IBM support.

## An IPsec tunnel cannot be established

An IPsec tunnel (encrypted connection) between the IKE daemons on a client LPAR and an accelerator cannot be established.

### Symptoms

Network traffic between the client LPAR and the accelerator remains unencrypted.

### Diagnosing the problem

Increase the value of `IkeSysLogLevel` to `255` in the configuration file of the IKE daemon on the client LPAR (file name: `iked.conf`) to obtain more detailed log messages. Save the file and refresh the configuration of the IKE daemon to let the changes take effect.

To display IPsec tunnels from a client LPAR to an accelerator, run the following command from a USS shell on the client LPAR or from an ssh terminal communicating with that LPAR. To display all IPsec tunnels from that LPAR:

```
ipsec -w display
```

To display all tunnels from a particular TCP/IP stack (named `TCPIP` in the example), run the following command. Use it if you run more than one TCP/IP stack from the same client LPAR:

```
ipsec -p TCPIP -y display
```

If you know the IP addresses that a tunnel is supposed to use, you can also use this command, which displays information about a particular tunnel from a particular TCP/IP stack :

```
ipsec -p TCPIP -t 10.101.14.25 10.101.12.47 tcp 21 0 out
```

In this last example, the command displays the tunnels from a stack named TCPIP with an IP address of 10.101.14.25, which connects to an accelerator with the wall-IP address 10.101.12.47 over the outbound TCP (source) port 21. The number 0 means that the destination port can be any port.

A working tunnel shows a ConnectionState of connected. If the status is different check, check the following log files to find out if the security certificates can be accessed:

- SYSLOG on the client LPAR
- nssd.log on the NSS server LPAR

## Resolving the problem

1. Try to solve the problem by using the information in the log (error) messages.
2. If you cannot find the cause of the error using the log messages, verify that the configuration steps in Steps for authorizing resources for NSS were followed correctly.
3. If this does not help, contact IBM support.

# Appendix A. Appendix: sample configuration of an AT/TLS connection

An AT/TLS connection must exist between the IKE daemon and the NSS daemon on the NSS server LPAR. Experience has shown that customer environments are quite diverse. It is therefore hardly possible to provide a sample configuration for each imaginable setup. The sample configuration shown here covers a setup with two different LPARs. Modify and extend this sample to make it work in your environment.

The first LPAR in the sample is called the client LPAR. Is is supposed to open encrypted IPsec tunnels to one or more accelerators. The other is called the NSS server LPAR. It runs the NSS daemon that provides access to the key ring. The key ring contains the certificates that are needed for the mutual authentication of the two LPARs.

## TTLS rules in the policy-agent configuration-file

To enable AT/TLS communication between your LPARs, must add the appropriate set of TTLS rules to the Policy Agent configuration-file. This configuration file is used by the Policy Agent of the NSS daemon. Hence you find it on the machine on which the NSS daemon is running. Here is the sample configuration for TTLS communication between your LPARs in the Policy Agent configuration file:

```
#####################################################################
#                                                                   #
# Network Security Services                                         #
#                                                                   #
#####################################################################
##
## CA generated policy statements for TTLSRule(s) in support of NSS function.
##

TTLSGroupAction              NSS~TLS~ON
{
   TTLSEnabled               On
   Trace 255
}
TTLSCipherParms              NSS~CIPHER~PARMS
{
   V3CipherSuites            TLS_RSA_WITH_3DES_EDE_CBC_SHA
}
TTLSEnvironmentAction        NSS~TLS~CLIENT~ENV
{
   HandShakeRole             Client
   TTLSCipherParmsRef        NSS~CIPHER~PARMS
   TTLSKeyRingParms
   {
      Keyring                NSSD/TLS_SHARED_RING
   }
   TTLSEnvironmentAdvancedParms
   {
        CertificateLabel IKED_ATTLS
        ClientAuthType Required
   }
}
## Rule from client DWB1 to server DWP1
TTLSRule                     NSS~TLS~CLIENT~1
{
```

```
   RemotePortRange           4159
   Direction                 Outbound
   TTLSGroupActionRef        NSS~TLS~ON
   TTLSEnvironmentActionRef  NSS~TLS~CLIENT~ENV
}
On the NSS Server LPAR "DWP1" we use the following additional rules for
the inbound AT/TLS connection

TTLSEnvironmentAction        NSS~TLS~SERVER~ENV
{
   HandShakeRole             Server
   TTLSCipherParmsRef        NSS~CIPHER~PARMS
   TTLSKeyRingParms
   {
      Keyring                NSSD/TLS_SHARED_RING
   }
   TTLSEnvironmentAdvancedParms
   {
        CertificateLabel NSSD_ATTLS
        ClientAuthType Required
   }
}
TTLSRule                     NSS~TLS~SERVER
{
   LocalPortRange            4159
   Direction                 Inbound
   TTLSGroupActionRef        NSS~TLS~ON
   TTLSEnvironmentActionRef  NSS~TLS~SERVER~ENV
}
```

Activate the changes by refreshing the IPsec Policy Agent configuration on the NSS server LPAR. For instructions, see step 2 on page 18 in "Activating the changed configuration of the Policy Agent" on page 18.

## Dedicated user to start the NSS daemon

You start a separate instance of the NSS daemon (called NSS started task) for the AT/TLS connection. A dedicated user (ID) for running this task is recommended. The following sample code generates a user ID with the name NSSD:

```
ADDUSER NSSD NAME('STARTED TASK')   DFLTGRP(OMVSGRP) +
        DATA('STARTED TASK NSSD')   NOPASSWORD       +
        OMVS(UID(0) HOME('/') PROGRAM('/bin/sh'))    +
        OWNER(STCGROUP)
...
RDEF STARTED NSSD*.**    STDATA(USER(NSSD)    GROUP(STCGROUP))
...
SETR REFRESH RACLIST(STARTED) GENCMD(*) GENERIC(*)
```

## Certificates

Both LPARs use a certificate to authenticate each other. It is recommended that you create a certificate that is different from the one used for the encryption of network traffic between an LPAR and an accelerator. However, you can store the certificate in the same key ring. In the following sample JCL job, however, a different key ring with the name TLS_SHARED_RING is used. The sample job generates a self-signed root certificate (CA), a certificate that authenticates the NSS server LPAR, and a certificate that authenticates the client LPAR (the LPAR that you want to configure for encrypted network traffic):

```
//* Create a CA certificate (self-signed in TLS_SHARED_RING)
  RACDCERT CERTAUTH GENCERT                                  +
    SUBJECTSDN(                                              +
      O('IBM DEUTSCHLAND RESEARCH && DEVELOPMENT GMBH')      +
```

```
      OU('Z SYSTEMS SW TESTLAB')                          +
      C('DE'))                                            +
    NOTBEFORE(DATE(2015-08-11))                           +
    NOTAFTER(DATE(2020-03-15))                            +
    KEYUSAGE(CERTSIGN)                                    +
    WITHLABEL('NSSD_CA1_ATTLS')

//* connect the CA to keyring TLS_SHARED_RING

  RACDCERT ID(NSSD) CONNECT(CERTAUTH LABEL('NSSD_CA1_ATTLS') +
           RING(TLS_SHARED_RING) USAGE(CERTAUTH))

//* CREATE A  CERTIFICATE FOR THE NSSD SERVER

  RACDCERT ID(NSSD) GENCERT                               +
      SUBJECTSDN(CN('BOEDWP1.BOEBLINGEN.DE.IBM.COM')      +
      O('IBM DEUTSCHLAND RESEARCH && DEVELOPMENT GMBH')   +
      OU('SYSTEM Z SW TESTLAB')                           +
      L('BOEBLINGEN') SP('BADEN WUERTTEMBERG') C('DE'))   +
      NOTAFTER(DATE(2020-03-15))                          +
      SIZE(2048)      WITHLABEL('NSSD_ATTLS')             +
      SIGNWITH(CERTAUTH LABEL('NSSD_CA1_ATTLS'))

  RACDCERT ID(NSSD) CONNECT(ID(NSSD) LABEL('NSSD_ATTLS') +
         RING(TLS_SHARED_RING)

//* Create a personal certificate for the IKED client.

  RACDCERT ID(NSSD) GENCERT                               +
      SUBJECTSDN(CN('BOEDWP1.BOEBLINGEN.DE.IBM.COM')      +
      O('IBM DEUTSCHLAND RESEARCH && DEVELOPMENT GMBH')   +
      OU('SYSTEM Z SW TESTLAB')                           +
      L('BOEBLINGEN') SP('BADEN WUERTTEMBERG') C('DE'))   +
      NOTAFTER(DATE(2020-03-15))                          +
      SIZE(2048)      WITHLABEL('IKED_ATTLS')             +
      SIGNWITH(CERTAUTH LABEL('NSSD_CA1_ATTLS'))

  RACDCERT ID(NSSD) CONNECT(ID(NSSD) LABEL('IKED_ATTLS') +
           RING(TLS_SHARED_RING)

  SETROPTS RACLIST(DIGTCERT) REFRESH
```

## RACF resources and permits

This configuration requires certain resources and permits in the RACF FACILITY class. How to create these is described in the IBM Knowledge Center under the heading:

Steps for authorizing resources for NSS

## RACF IRR.DIGTCERT FACILITY resource

The IRR.DIGTCERT FACILITY resource in RACF is used in the following way to control access to the **RACDCERT** command:

```
RDEFINE FACILITY   IRR.DIGTCERT.ADD        UACC(NONE)
RDEFINE FACILITY   IRR.DIGTCERT.ADDRING    UACC(NONE)
RDEFINE FACILITY   IRR.DIGTCERT.CONNECT    UACC(NONE)
RDEFINE FACILITY   IRR.DIGTCERT.GENCERT    UACC(NONE)
RDEFINE FACILITY   IRR.DIGTCERT.GENREQ     UACC(NONE)
RDEF    FACILITY   IRR.DIGTCERT.EXPORT     UACC(NONE)
RDEF    FACILITY   IRR.DIGTCERT.EXPORTKEY  UACC(NONE)
RDEF    FACILITY   IRR.DIGTCERT.DELETE     UACC(NONE)
RDEF    FACILITY   IRR.DIGTCERT.LIST       UACC(NONE)
...
```

```
PE IRR.DIGTCERT.ADD       CL(FACILITY) ID(IBMGRP01)  ACC(CONTROL)
PE IRR.DIGTCERT.ADDRING   CL(FACILITY) ID(IBMGRP01)  ACC(UPDATE )
PE IRR.DIGTCERT.CONNECT   CL(FACILITY) ID(IBMGRP01)  ACC(CONTROL)
PE IRR.DIGTCERT.GENCERT   CL(FACILITY) ID(IBMGRP01)  ACC(CONTROL)
PE IRR.DIGTCERT.GENREQ    CL(FACILITY) ID(IBMGRP01)  ACC(CONTROL)
PE IRR.DIGTCERT.LIST      CL(FACILITY) ID(IBMGRP01)  ACC(CONTROL)
PE IRR.DIGTCERT.LISTRING  CL(FACILITY) ID(IBMGRP01)  ACC(UPDATE)
PE IRR.DIGTCERT.EXPORT    CL(FACILITY) ID(IBMGRP01)  ACC(CONTROL)
PE IRR.DIGTCERT.EXPORTKEY CL(FACILITY) ID(IBMGRP01)  ACC(CONTROL)
PE IRR.DIGTCERT.DELETE    CL(FACILITY) ID(IBMGRP01)  ACC(CONTROL)
...
PE IRR.DIGTCERT.LIST      CL(FACILITY) ID(NSSD) ACC(READ)
PE IRR.DIGTCERT.LISTRING  CL(FACILITY) ID(NSSD) ACC(READ)
SETROPTS RACLIST(FACILITY) REFRESH
...
PERMIT CSF*    CLASS(CSFSERV) ID(NSSD) ACCESS(READ)
SETROPTS RACLIST(CSFSERV) REFRESH
...
RACDCERT ID(NSSD   ) ADDRING(TLS_SHARED_RING)
```

For more information, see the following chapter in the IBM Knowledge Center:

RACDCERT (Manage RACF digital certificates)

## IKE daemon configuration file on the client LPAR

When an AT/TLS connection is established, the NSS server (daemon) communicates with the IKE daemon on the client LPAR. To enable this type of communication, the IKE daemon must be configured accordingly. You do this in a file called `iked.conf` on the client LPAR (the LPAR that you want to configure for encrypted network traffic with one or more accelerators). The following values are used in the sample configuration of the IKE daemon:

**9.152.87.129**
: The IP address by which the NSS server (running the NSS daemon) can be reached.

**DWB1_TCPIP**
: The client name that identifies the IKE daemon on the client LPAR when connecting to the NSS daemon on the NSS server LPAR.

**DWB1IP1**
: A user ID that exists on the NSS server LPAR and that the IKE daemon (running on the client LPAR) uses to connect to the NSS server LPAR.

The sample configuration as part of the `iked.conf` file on the client LPAR:

```
NetworkSecurityServer 9.152.87.129 Port 4159
Identity X500dn "CN=NSSD,OU=SYSTEM Z SW TESTLAB,
O=IBM DEUTSCHLAND RESEARCH & DEVELOPMENT GMBH,L=BOEBLINGEN,
SP=BADEN WUERTTEMBERG,C=DE"
}
...
# NssStackConfig    stackname         (dynamically modifiable)
# Used to configure a stack as a Network Security client.
NssStackConfig TCPIP
{

# ClientName        clientname        (dynamically modifiable)
# Specifies the Network Security client name for the stack.
  ClientName  DWB1_TCPIP

# ServiceType       Cert,RemoteMgmt   (dynamically modifiable)
# Specifies the types of centralized services requested
```

```
# from the Network Security server.
  ServiceType Cert

# ServiceType       Cert,RemoteMgmt   (dynamically modifiable)
# Specifies the types of centralized services requested
# from the Network Security server.
  ServiceType RemoteMgmt

# UserId            userid            (dynamically modifiable)
# Specifies the RACF userid that will be used to verify access
# for this stack to the services provided by the
# Network Security server.
  UserId      DWB1IP1

# AuthBy      Password pw, Passticket (dynamically modifiable)
# Specifies the mechanism by which the Network Security server
# should authenticate the client TCPIP stack.
  AuthBy      Password secret
}
```

# Appendix B. Appendix: terms and acronyms in this document

Read a brief description of the terms and acronyms that are used in this document.

**Defense Manager (DM) daemon**
On z/OS, the component or service that is responsible for defensive filtering.

**Internet Key Exchange (IKE) daemon**
On z/OS, the Internet Key Exchange daemon is responsible for retrieving the IP security policy from the Policy Agent, and for dynamically managing keys that are associated with dynamic networking tunnels. The IKE daemon implements IPsec protocols for the dynamic creation of security associations (SAs) with peer components that also support these protocols. It can also automate the management of cryptographic keys, that is, their creation, distribution, and maintenance.

**Network Security Services (NSS)**
A set of libraries for the development of security-enabled applications. NSS contains cryptographic libraries that support Transport Layer Security (TLS), Secure Sockets Layer (SSL), and S/MIME.

**Network Security Services (NSS) daemon (z/OS)**
A continuously running z/OS program that provides IPsec services and resources, such as authentication services and certificates.

**Network security services (NSS) database (Linux)**
A Netezza database on an accelerator that provides resources, such as security certificates, for IPsec services.

**Policy Agent**
A z/OS component that can take on various roles in connection with IP networking. The Policy Agent reads and parses policy definitions (=sets of rules) and makes these policies available to TCP/IP stacks or policy clients. The Policy Agent can be configured to start, stop, monitor, or restart dependent components, such as the Internet Key Exchange (IKE) daemon, the NSS daemon, the SYSLOG daemon, or the Traffic Regulation Management (TRM) daemon.

**SYSLOG daemon**
The SYSLOG daemon handles the logging of all other events that are not handled by the TRM daemon. It also controls where the log messages are written. The SYSLOG provides valuable information for troubleshooting.

**TCP/IP stack**
On z/OS, an addressable (named) unit or component that is responsible for the processing of network packages according to the TCP/IP set of protocols. The term is often used in a different sense in other publications, in which it means the TCP/IP set of protocols.

**Traffic Regulation Management (TRM) daemon**
The Traffic Regulation Management (TRM) daemon on z/OS can be viewed as a message writer that selects and logs certain events in the SYSLOG. These are events that were caused by IPsec services as well as by the implementation of policies for traffic regulation or intrusion detection services (IDS).

You can look up terms and acronyms that are not listed here on the IBM Terminology website.

# Index

# Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing*
*IBM Corporation*
*North Castle Drive, MD-NC119*
*Armonk, NY 10504-1785*
*US*

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

*Intellectual Property Licensing*
*Legal and Intellectual Property Law*
*IBM Japan Ltd.*
*19-21, Nihonbashi-Hakozakicho, Chuo-ku*
*Tokyo 103-8510, Japan*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those

websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

*IBM Director of Licensing*
*IBM Corporation*
*North Castle Drive, MD-NC119*
*Armonk, NY 10504-1785*
*US*

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to

IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year).
Portions of this code are derived from IBM Corp. Sample Programs.
© Copyright IBM Corp. _enter the year or years_.

## Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.


Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Netezza and NPS are trademarks or registered trademarks of IBM International Group B.V., an IBM Company.

UNIX is a registered trademark of The Open Group in the United States and other countries.

## Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

### Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

## Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

## Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

## Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Product Number: 5697-DA5

SH12-7077-01